DOCUMENT RESUME

ED 422 359 TM 028 911

AUTHOR Yan, Duanli; Lewis, Charles; Stocking, Martha
TITLE Adaptive Testing without IRT.
PUB DATE 1998-04-00
NOTE 23p.; Paper presented at the Annual Meeting of the National Council on Measurement in Education (San Diego, CA, April 12-16, 1998).
PUB TYPE Reports - Evaluative (142) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Adaptive Testing; Algorithms; *Computer Assisted Testing; *Item Response Theory; Models; *Nonparametric Statistics; *Regression (Statistics); Simulation; *Test Construction

ABSTRACT

It is unrealistic to suppose that standard item response theory (IRT) models will be appropriate for all new and currently considered computer-based tests. In addition to developing new models, researchers will need to give some attention to the possibility of constructing and analyzing new tests without the aid of strong models. Computerized adaptive testing currently relies heavily on IRT. Alternative, empirically based, nonparametric adaptive testing algorithms exist, but their properties are little known. This paper introduces an adaptive testing algorithm that balances maximum differentiation among test takers with stable estimation at each stage of testing, and compares this algorithm with a traditional one using IRT and maximum information. The adaptive testing algorithm introduced is based on the classification and regression tree approach described in L. Breiman, J. Friedman, R. Olshen, and C. Stone (1984) and J. Chambers and T. Hastie (1992). Simulation results from the regression tree approach were compared with simulation results from three parameter logistic model IRT. Simulation results show that the nonparametric tree-based approach to adaptive testing may be superior to conventional IRT-based adaptive testing in cases where the IRT assumptions are not satisfied. It clearly outperformed one-dimensional IRT when the pool was strongly two-dimensional. A technical appendix describes the algorithm. (Contains three figures and six references.) (SLD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)
☑ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Adaptive Testing without IRT

Duanli Yan, Charles Lewis and Martha Stocking
Educational Testing Service
Princeton, NJ

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY
Duanli Yan
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

ED 422 359

It is unrealistic to suppose that standard item response (IRT) models will be appropriate for all the new and currently considered computer-based tests . In addition to developing new models, we also need to give some attention to the possibility of constructing and analyzing new tests without the aid of strong models. Computerized adaptive testing currently relies heavily on item response theory. Alternative, empirically based, non-parametric adaptive testing algorithms exist, but their properties are little known.

Introduction

Wainer *et al.* (1991) and Wainer *et al.* (1992) introduced a testlet-based algebra exam and compared a hierarchically constructed (adaptive) 4-item testlet with a linear (fixed format) testlet under various conditions. They compared an adaptive test using an optimal 4-item tree and a best fixed 4-item test (both defined in terms of maximum differentiation) through cross-validation, and found that overall the adaptive testlet dominates the best fixed testlet, but the superiority (at a considerable cost for adaptive testlet over fixed testlet) is modest. They also found that the adaptive test outperforms the fixed test for the groups with extreme observed scores. They concluded that, for circumstances similar to their cases, "a fixed format testlet that uses the best items in the pool can do almost as well as the optimal adaptive testlet of equal length from that same pool."

TM028911

Paper presented at the annual meeting of the National Council on Measurement in Education, San Diego, CA, April 1998.

Schnipke and Green (1995) compared an item selection algorithm based on maximum differentiation among test takers with one using item response theory and based on maximum information. Overall, adaptive tests based on maximum information provided the most information over the widest range of ability values and, in general, differentiated among test takers slightly better than the other tests. Although the maximum differentiation technique may be adequate in some circumstances, adaptive tests based on maximum information were clearly superior in their study.

This paper introduces an adaptive testing algorithm that balances maximum differentiation among test takers with stable estimation at each stage of testing, and compares this algorithm with a traditional one using item response theory and maximum information.

Method

In this paper, we consider adaptive testing as a prediction system. Specifically, we use adaptive testing to predict the observed scores that test takers would have received if they had taken every item in a reference test or a pool. (We restrict our attention to binary items, scored correct or incorrect.) This is a non-parametric approach in the sense that we do not introduce latent traits or true scores. We are only considering the observed number-correct scores test takers would have received if they had taken every item we could have given. In other words, our criterion is the total observed score for a pool or reference test. The adaptive testing algorithm we introduce in this paper is based on the classification and regression tree approach described in Breiman, Friedman, Olshen and Stone (1984) and in Chambers and Hastie (1992).

In order to construct an adaptive test as a prediction system, we need to have a calibration sample. Specifically, we need a sample of test takers who take every item in

the pool that will be used for adaptive testing. (For operational use, incomplete calibration designs would obviously be necessary.) We can then compute the criterion (total observed score) for these test takers. This is analogous to the calibration sample one needs when using IRT to do adaptive testing. However, the purpose of the IRT calibration sample is to calibrate items. Our purpose is not to calibrate items individually, but to generate a regression tree.

Figure 1 is an example of such a regression tree. The vertical axis represents the stage of testing and the horizontal axis identifies the prediction of the total score at each stage. In this example, there are 9 stages (*i.e.*, each test taker would be administered 9 items). The nodes of the tree are plotted as octagons with item numbers inside. The branches represent the paths test takers could follow in the test, taking the right branch after answering the item in the octagon correctly and the left branch after answering incorrectly. At the end, the locations of the terminal nodes, or leaf nodes, give the final predictions of test takers' total scores.

Once the regression tree has been constructed (and validated -- see below), it may be used to administer an adaptive test. Thus, based on Figure 1, all test takers would first be administered item 31. Test takers answering correctly would receive item 27. Those answering incorrectly would get item 28. Test takers continue through the tree to the terminal nodes and receive the corresponding final predicted total score as their score on the test. For instance, test takers who receive item 5 as the last item, and answer it correctly, would have a predicted total score of 32.5.

Returning to the construction of the tree, suppose we have a calibration sample of test takers who answered every item in a pool of items. The total number of correct responses for each test taker is the criterion we will use. Our regression tree begins with the item (in Figure 1, item 31) that best predicts the observed score in a least squares

sense for these test takers. It splits the calibration sample into two subsamples: those test takers who answered the item incorrectly and those who answered correctly. They are represented as nodes in Figure 1: the left node with number 28 inside and right node with number 27 inside. These two subsamples have maximum differentiation between them (*i.e.*, maximum sum of squares between subsamples). The horizontal locations of the nodes are the mean total scores for these two sub-samples.

The construction of the tree continues by finding the best predicting item for those test takers responding correctly to the first item (in Figure 1, item 27), as well as the best item for those with an incorrect response to the first item (in Figure 1, item 28). At each stage, the total calibration sample is split into subsamples and an optimal item is chosen for each subsample. At each stage, it is also the case that subsamples with similar average criterion scores are combined as the tree progresses to keep the total number of such subsamples within reasonable limits. In Figure 1, the nodes with test takers who correctly answered item 28 and test takers who incorrectly answered item 27 are combined, and the combined subsamples are administered item 16. At the end of the process, the adaptive test score given to each test taker is the average criterion score for the final subsample to which the test taker has been classified (in Figure 1, the combined leaf nodes).

A portion of the prediction for the calibration sample capitalizes on chance. To evaluate the procedure, we construct the regression tree in a calibration sample and apply the predictions from the calibration sample to compare with the observed scores in an application sample. This application sample has the same structure as the calibration sample. In other words, every test taker answers every item, so a criterion observed score can be computed. The precision of estimation using the regression tree as an adaptive test may be measured using the mean of the squared discrepancies (or residuals) between predicted and observed scores in the application sample. For purposes of interpretation,

this quantity may be compared to the variance of the observed scores in the application sample. In particular, we will consider the proportion of variance accounted for by the tree-based predictions.

Results

We compared simulation results from the regression tree approach with simulations using 3PL IRT and maximum information for the same situations. For our first set of simulations, we constructed our calibration sample using the 3PL IRT model to generate item responses for a sample of 500 simulated test takers to 494 items in an actual item pool for an operational computer adaptive test assessing quantitative reasoning. (Specifically, we used the 3PL IRT model with item parameters set equal to the estimates from the operational pool.)

We constructed a regression tree as described in the method section for a 19-item adaptive test for this calibration sample. The mean of the squared residuals between predictions and total observed scores for the calibration sample is 11.7. This quantity may be compared to the variance of the total observed scores for the sample, or 6,586.0. Thus 99.8% of the total observed score variance is accounted for in the calibration sample using the predictions from the regression tree.

Next, we used the regression tree predictions based on the calibration sample to compare with the total (IRT-based) *true* scores (rather than total observed scores) in an application sample of size 10,000, constructed in the same way as the calibration sample. The mean squared residual in the application sample, based on the calibration predictions at the end of the 19-item test is 1223.7 (with original true score variance of 6,457.5), which means the predictions account for 81.0% of the total true score variance. From this result, we see that there was substantial capitalization on chance in the calibration sample.

From the same calibration sample we used to construct the regression tree, we also obtained 3PL item parameter estimates using PARSCALE (Muraki & Bock, 1993). We then carried out an IRT-based (maximum information) adaptive testing simulation on the application sample using these estimated item parameters. As estimates of total true score, we used maximum likelihood estimates of the latent trait, transformed using the test characteristic curve for the entire pool. The mean squared residual between these estimates and the total true scores in the application sample is 514.5. Comparing this with the total true score variance for the application sample, we see that the IRT-based estimates account for 92.0% of that variance, substantially more than when using the tree-based predictions. Figure 2 provides a more detailed comparison of the regression tree and IRT-based CATs as a function of test length.

For our second set of simulations, we considered what would happen if the 3PL IRT assumptions were violated. We used the same pool, but split it into two equal parts such that half of the items were considered to measure one latent trait and the other half measured a second, uncorrelated, latent trait. The parameters for all items were left unchanged. A calibration sample consisting the responses to all items in the pool for a sample of 500 simulated test takers was generated, based on the two dimensional latent trait model just described. Specifically, for each simulated test taker, two latent trait values were sampled. One of these was used as the basis for response generation for items in the first half of the pool, while the other was used to generate responses to items in the second half of the pool. We used the resulting data as our calibration sample for both the tree-based approach and the one-dimensional IRT model, as before. (Note that the 3PL model was fit simultaneously to all items in the pool, ignoring which half they were in.)

We also generated responses to all items in the pool for a new sample of 10,000 simulated test takers in the manner just described for use as our application sample. We

used the tree-based predictions from the calibration sample in the application sample for the regression tree approach. We also carried out a (one-dimensional) IRT adaptive testing simulation for this application sample, using the item parameters obtained from the calibration sample. The two-dimensional, IRT-based total true scores for the pool served as the evaluation criterion for both procedures. Specifically, we compared the mean squared residuals obtained for the two methods.

Fitting a regression tree to the data for the calibration sample produced a mean squared residual of 20.8, compared with a total observed score variance of 3,025.9. In other words, 99.3% of the observed score variance can be accounted for in this calibration sample using the predictions from the regression tree. (Note that the total observed score variance in this sample is much smaller than that obtained in the calibration sample based on the one dimensional IRT model: 3,025.9 compared to 6,586.0. This is a result of the fact that the between-set item correlations in our second design are all zero.)

In the application sample, using the tree-based predictions from the calibration sample to predict the total true scores gave a mean squared residual of 1,187.0. The total true score variance in the application sample is 3,180.0, so 62.7% of this variance is accounted for by the tree-based predictions. (Here we see an even more substantial chance capitalization than in the one-dimensional case.) True score estimates based on a 3PL IRT CAT produced a mean squared residual of 1,960.0 in the application sample, so that only 38.4% of the total true score variance is accounted for by these estimates. Figure 3 provides a more detailed comparison of the regression tree and IRT-based CATs as a function of test length.

Discussion

For our one-dimensional example, Figure 2 shows that, once the adaptive test is long enough, the IRT-based CAT produces consistently better estimates of true scores than does the tree-based approach, However, it is worth noting that, in the early stages of testing, the maximum likelihood estimates from the IRT-based CAT are very poor compared to those from the regression tree. This suggests a possible hybrid algorithm, using a regression tree to select the first few items on an adaptive test, and then switching to a maximum information IRT-based algorithm. This leaves open the question of how best to make the transition from regression tree to maximum likelihood estimates.

In our two-dimensional example, we see from Figure 3 that the regression tree clearly provides better prediction than the IRT-based CAT at all test lengths. However, it should be noted that our example is based on an extreme version of a two-dimensional model in which every item measures either one or the other dimension (but not both), and the two uncorrelated dimensions are taken to be equally important.

One of the limitations of the tree-based approach described in this paper is that there is no control of item exposure rates. (For instance, our algorithm now has everyone take the same first item.) Another limitation is that no attempt is made to control the content of the adaptive tests. A third limitation is that all test takers in the calibration and application samples were assumed to have answered all items in the pool. (All these limitations also apply to the IRT-based algorithm we used for comparison purposes in this study. It should be noted, however, that operational IRT CATs have none of these limitations.) Future research will address these and related issues.

Conclusions

We have developed a non-parametric, tree-based approach to adaptive testing and shown that it may be superior to conventional, IRT-based adaptive testing in cases where the IRT assumptions are not satisfied. In particular, we showed that the tree-based approach clearly out-performed (one-dimensional) IRT when the pool was strongly two-dimensional.

Educational Importance

With new types of computer-based tests being considered, it is important to have new psychometric tools available to be used with these tests. The non-parametric, tree-based approach to adaptive testing described here is one such tool. We expect it to be particularly useful when items test more complex domains than is currently the case with IRT-based adaptive testing.

Technical Appendix - Description of algorithm

Our regression trees are constructed as follows. For each node, we select an unused item that gives the maximum differentiation (in a least squares sense) on the criterion score for splitting the current node into two nodes. For each stage, we compare all the nodes at that stage by computing the pairwise t-statistics and effect size measures, using the criterion score. If, for some pair of nodes, the absolute value of the t-statistic is less then some pre-set critical value, or the absolute value of the effect size measure is less than some pre-set critical value, then we combine the two nodes. (If there is more than one pair of nodes that meets either of these criteria, we start by combining the pair with the smallest t-statistic (or smallest effect size if no t-statistic is less than the critical value). We then recompute all t-statistics and effect sizes for this new node with the others and repeat the process until all pairs of nodes are distinct in terms of their t-statistics and effect sizes.

We continue constructing the regression tree stage by stage in this manner until a specified fixed test length is reached. At the final stage, each test taker in a sample is classified to one of the leaf nodes based on matching their response pattern with the regression tree structure. The prediction of that individual's criterion score is the average score of the leaf node in the regression tree to which the individual is classified.

Exhibit A reproduces an edited version of a portion of the output from the program we use to construct regression trees. Specifically, it is the output describing the construction of the tree illustrated in Figure 1. The information given in line 007 describes the complete calibration sample (node 0) as having 250 (simulated) test takers, a mean criterion score of 34.7360 and a sum of squared deviations of individual scores around this mean (Deviance) of 28608.5760. Line 012 repeats some of this information, and notes

that item 31 has been selected as the first item in the tree. The remaining output in lines 015 - 023 will be of more interest at later stages.

Lines 027 and 028 describe nodes 1 and 2, defined as those test takers who answer item 31 incorrectly or correctly, respectively. Specifically, there are 71 of the former and 179 of the latter, with mean criterion scores of 25.3521 and 38.4581, respectively. Lines 029 and 030 give the t-statistic and effect size measure used to compare nodes 1 and 2. Both values exceed their respective criteria, so no combining of nodes occurs at this stage. Lines 035 and 036 indicate that items 28 and 27 have been chosen for nodes 1 and 2, respectively. Line 039 gives the total within-node sum of squares at stage 1 as 19876.6329. Note that this is the sum of the sums of squares for each of the two nodes at this stage. Line 041 gives the proportion of variance accounted for at this stage, obtained by subtracting the ratio of the deviance at this stage to the deviance at stage 0 from unity. Lines 047 and 048 report the standard deviations for nodes 1 and 2.

Lines 052 - 055 describe the four nodes at stage 2 defined by incorrect and correct answers to items 28 and 27. Lines 056 - 062 give all pairwise comparisons for the nodes at this stage, as well as the comparison with the smallest t-statistic (obtained for nodes 4 and 5). Since this value (-1.2691) is less in absolute value than our critical value of 2.0, these two nodes are combined. The new nodes are described in lines 068 - 070, and the comparisons are given in lines 073-076. No further combination is indicated, so items are chosen for each of these nodes (2, 16, and 33, respectively), and the final description is given in lines 078 - 095. The actual output continues in this fashion until the specified number of stages (test length) has been reached.

References

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees.* Pacific Grove, CA: Wadsworth & Brooks/Cole Advanced Books & Software.

Chambers, J. M., & Hastie, T. J. (1992). *Statistical models.*. Pacific Grove, CA: Wadsworth & Brooks/Cole Advanced Books & Software.

Muraki, E., & Bock, D. (1993). *PARSCALE: IRT based test scoring and item analysis for graded open-ended exercises and performance tasks.* [Computer program]. Chicago, IL: Scientific Software, Inc.

Schnipke, D., & Green, B. (1995). A comparison of item selection routines in linear and adaptive tests. *Journal of Educational Measurement, 32,* 227-242.

Wainer, H., Lewis, C., Kaplan, B., & Braswell, J. (1991). Building algebra testlets: A comparison of hierarchical and linear structures. *Journal of Educational Measurement, 28,* 311-324.

Wainer, H., Kaplan, B., & Lewis, C. (1992). A comparison of the performance of simulated hierarchical and linear testlets. *Journal of Educational Measurement, 29,* 243-251.

Exhibit A. Sample Output from Program to Construct Regression Trees

```
001  ADAPTIVE TESTING WITHOUT IRT - Calibration Sample Tree Table
002
003
004
005  ---------------------- Stage = 0 -------------------
006
007  NodeInfo:istage=   0 inode=   0 Nsubj= 250 yval= 34.7360 Dev=28608.5760
008
009
010  Stage Node  Item  Parents              N     Deviance       Mean      T          E
011
012     0    0   31   -9,-9,-9,-9,-9      250   28608.5760    34.7360   999.0000  999.0000
013
014
015  Total Deviance at Stage    0:    28608.5760
016
017  Proportion of Variance Accounted For:    0.0000
018
019  Standard Deviation at root:   10.6974
020
021  Standard Deviation for Each Node at This Stage:
022
023  Node =    0   SD =    10.6974
024
025  ---------------------- Stage = 1 -------------------
026
027  NodeInfo:istage=   1 inode=   1 Nsubj=  71 yval= 25.3521 Dev=3716.1972
028  NodeInfo:istage=   1 inode=   2 Nsubj= 179 yval= 38.4581 Dev=16160.4358
029  In Combine:inode,jnode,t,e =   1    2   -11.6993    -1.5452
030       Inode,Jnode,tmin,emin =   1    2   -11.6993    -1.5452
031
032
033  Stage Node  Item  Parents              N     Deviance       Mean      T          E
034
035     1    1   28    0,-9,-9,-9,-9       71    3716.1972    25.3521   -11.6993    1.5452
036     1    2   27    0,-9,-9,-9,-9      179   16160.4358    38.4581   999.0000  999.0000
037
038
```

Exhibit A. Sample Output from Program to Construct Regression Trees (Continued)

```
039  Total Deviance at Stage    1:    19876.6329
040
041  Proportion of Variance Accounted For:    0.3052
042
043  Standard Deviation at root:   10.6974
044
045  Standard Deviation for Each Node at This Stage:
046
047  Node =    1   SD =     7.2347
048  Node =    2   SD =     9.5017
049
050  ---------------------- Stage = 2 -------------------
051
052  NodeInfo:istage=   2 inode=   3 Nsubj=  44 yval= 22.8409 Dev=1069.8864
053  NodeInfo:istage=   2 inode=   4 Nsubj=  27 yval= 29.4444 Dev=1916.6667
054  NodeInfo:istage=   2 inode=   5 Nsubj=  70 yval= 31.7857 Dev=3251.7857
055  NodeInfo:istage=   2 inode=   6 Nsubj= 109 yval= 42.7431 Dev=7790.8073
056  In Combine:inode,jnode,t,e =   3    4    -3.6375    -0.9405
057  In Combine:inode,jnode,t,e =   3    5    -8.0368    -1.4907
058  In Combine:inode,jnode,t,e =   3    6   -17.9650    -2.8575
059  In Combine:inode,jnode,t,e =   4    5    -1.2691    -0.3012
060  In Combine:inode,jnode,t,e =   4    6    -7.2206    -1.5573
061  In Combine:inode,jnode,t,e =   5    6    -9.4833    -1.4190
062       Inode,Jnode,tmin,emin =   4    5    -1.2691    -0.3012
063
064  Estimations during combination:
065
066  Stage Node  Item  Parents              N     Deviance       Mean      T          E
067
068     2    3         1,-9,-9,-9,-9        44    1069.8864    22.8409   999.0000  999.0000
069     2    4         1, 2,-9,-9,-9        97    5275.2577    31.1340    -1.2691    0.3012
070     2    5         2,-9,-9,-9,-9       109    7790.8073    42.7431   999.0000  999.0000
071
072
073  In Combine:inode,jnode,t,e =   3    4    -7.7947    -1.3126
074  In Combine:inode,jnode,t,e =   3    5   -17.9650    -2.8575
075  In Combine:inode,jnode,t,e =   4    5   -10.4748    -1.4563
076       Inode,Jnode,tmin,emin =   3    4    -7.7947    -1.3126
077.
```

Exhibit A. Sample Output from Program to Construct Regression Trees (Continued)

```
078   Stage Node  Item  Parents              N     Deviance       Mean      T          E
079
080      2    3    2    1,-9,-9,-9,-9       44    1069.8864    22.8409   -7.7947    1.3126
081      2    4   16    1, 2,-9,-9,-9       97    5275.2577    31.1340  999.0000  999.0000
082      2    5   33    2,-9,-9,-9,-9      109    7790.8073    42.7431  999.0000  999.0000
083
084
085   Total Deviance at Stage    2:    14135.9514
086
087   Proportion of Variance Accounted For:    0.5059
088
089   Standard Deviation at root:   10.6974
090
091   Standard Deviation for Each Node at This Stage:
092
093   Node =    3   SD =     4.9311
094   Node =    4   SD =     7.3746
095   Node =    5   SD =     8.4543
```

Figure 1. Regression Tree Structure

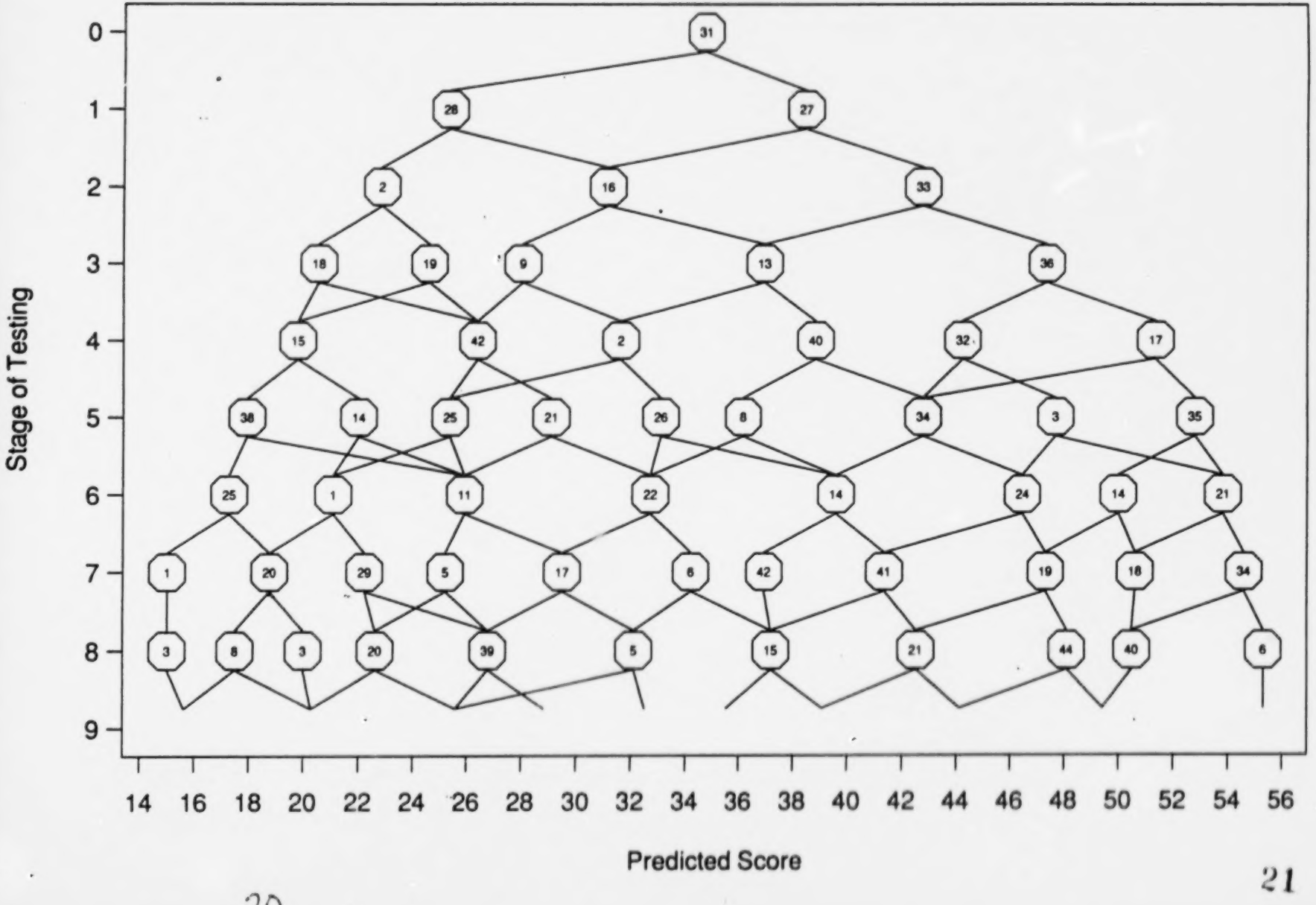

Figure 2

Comparison of Tree-based and IRT CATs in One-dimensional Application Sample
(referring to true scores)
I = IRT, T = Tree

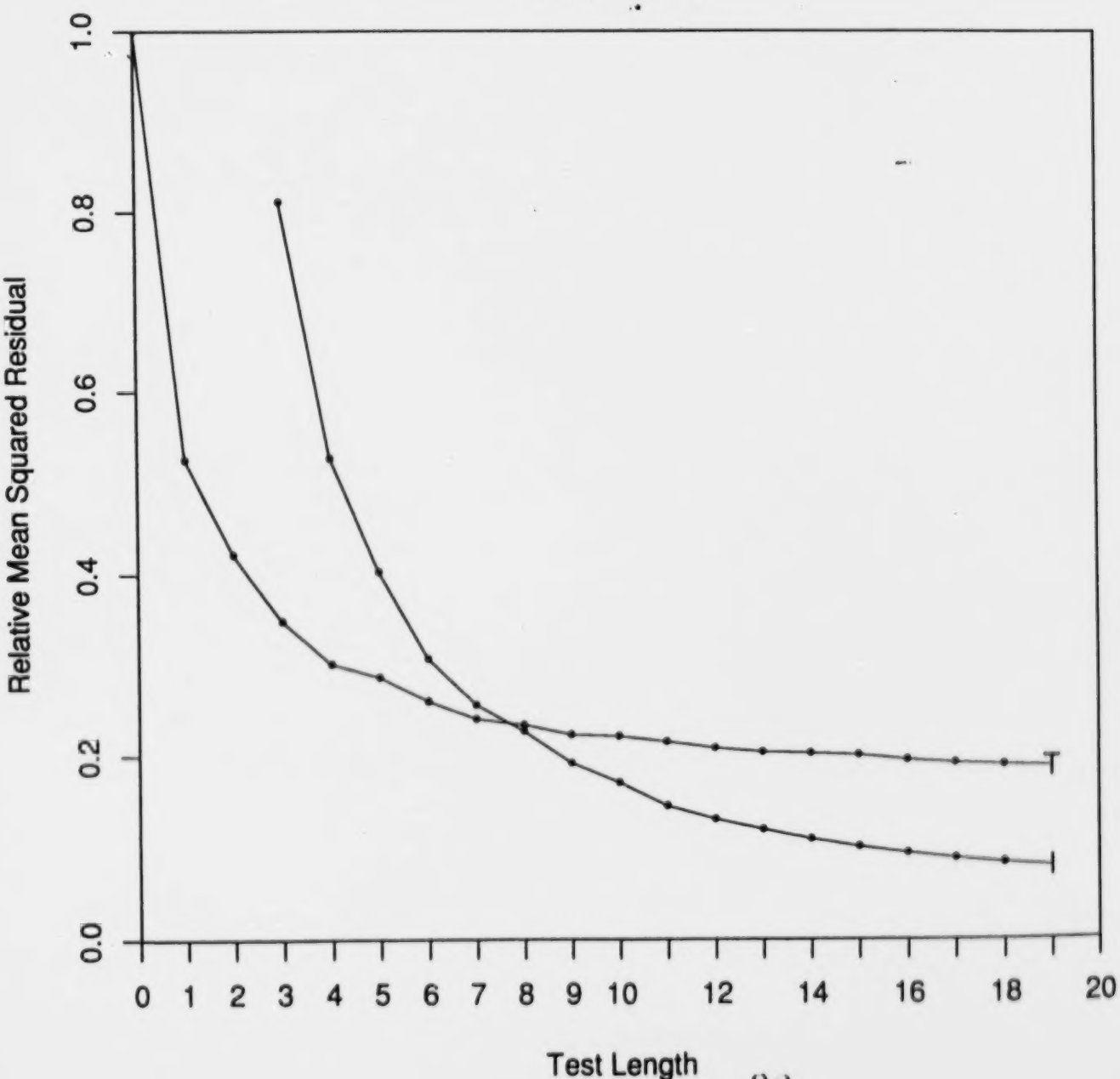

Figure 3

Comparison of Tree-based and IRT CATs in Two-dimensional Application Sample
(referring to true scores)
I = IRT, T = Tree